Final filtered output

Max-pooling Maximum of all 4 sum values

Make the filtered output C0 as 0

Shift Sum by the shift amount

Sum of all Partial products

If Sum < 0

Image 128-bit

(Always positive)

Range (0, 255)

Image 3x3 \* Kernel 3x3

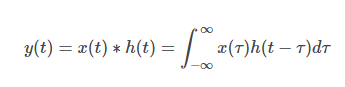
Kernel 72-bit

(Can be negative)

Range (-127, 128)

**Convolution:**

* Convolution is an important mathematical operation that is widely used in signal and image processing. The process of convolution comprises two 1-D input values.
* In our case, the 512x512 image is our input signal that needs to be filtered. The 3x3 kernel is the filter value with which the image will be filtered.
* It represents the weighted sum of input image and convolution kernel matrix.

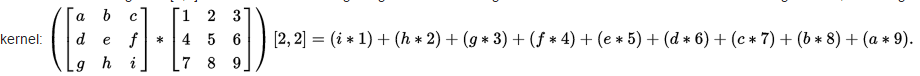


**Convolution Kernel:**

* Convolution kernel is a small matrix that is used to sharpen, blur, emboss or detect edges for a given input image. In our case, a sharp-kernel is used to sharpen the input image –lena.png.

**Why Convolution in Image processing:**

* We are performing convolution on the given image in order to enhance the core image with respect to the corner pixels.
* Upon convolution with the kernel the image is filtered and the center pixel values are matched with the corner pixels.
* Image convolution is used in edge detection, sharpening, blurring, enhancing, embossing input image pixels.



**Operation:**

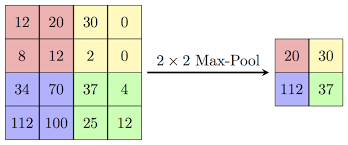
* At each cycle, a 4x4 sub-matrix, which is part of Image, 3x3 kernel and a -2-bit shift amount are latched to the design as input.
* The Image is split into four 3x3 matrices by shifting row-wise and column-wise.
* These four sub-matrices are convolved with the 3x3 kernel to produce 9 partial products in a parallel manner. Meaning, for all 4 matrices, convolution happens in parallel.
* We have to keep in mind that the image has pixel values as the contents of the matrix.
* Hence, image matrix only contains positive values in the range of 0 to 255.
* But, the kernel matrix can have negative values and with a range of -127 to 128.
* Convolution involves multiplication of input signal and filter followed by adding all the values. All operation needs to be signed.
* Once the final sum of 9 partial products is calculated, we need to divide the sum value by 2 to the power of shift amount. We could just right shift the sum value by the shift amount.
* If sum value is less than 0, then we make the sum value as zero. If sum value is greater than 0, we make sum as 255.

**New Approach for Floor/Ceil and Shift:**

* Consider a worst case negative number A. When we right shift A, we get another negative number. After convolution, the sum values are limited to the range of pixel values.
* Hence, it is unnecessary to shift the negative values and positive values.
* Instead, if design sees a negative value, it floors it to the minimum value and only for positive values shifting operation takes place.
* Hence, in my design, I am first checking if the sum is less than zero,
  + If sum less than zero, I will make it zero.
  + Else, I will right shift the sum by shift amount.
* Once, shift is done, I will check
  + If shifted SUM is greater than 255, then, I will make it 255.
  + Else, I will keep the last 8 bits of the shifted SUM.

**Max-Pooling:**

* Pooling layer is frequently used in convolutional neural networks with the purpose to progressively reduce the spatial size of the representation to reduce the amount of features and the computational complexity of the network.
* The most commonly used pooling layer is the MAXPOOL layer, with is provided by all deep learning libraries.
* Max-pooling is a sample-based discretization process.
* Its main objective is to down-sample the input signal/ image by reducing its dimensionality and allowing for assumptions to make about the features contained in the sub-regions behind.



**Why Max-pooling in Image processing:**

* Basically a max-pool of 2 x 2 would cause a filter of 2 by 2 to traverse over the entire matrix and pick the largest element from the window to be included in the next representation map.
* By performing, max-pooling, we are reducing image dimensions by not losing critical information.

**Operation in design:**

* Max pooling is done by applying a max filter to non-overlapping sub-regions of the initial representation.
* By the end of convolution, we will successfully, get 4 sum values for each image sub-matrix convolution.
* These 4 values are then, compared with each other and finally, the maximum of the 4 values is chosen and written in the output file.